TaxaMiner: An Experimentation Framework for Automated Taxonomy Bootstrapping

V Kashyap[1], C Ramakrishnan[2], C Thomas[2], D Bassu[3], T C Rindflesch[1] and A Sheth[2]

[1]LHNCBC, National Library of Medicine, 8600 Rockville Pike, Bethesda, MD 20894
[2]LSDIS Lab, Department of CS, University of Georgia, 415 GSRC, Athens, GA 30602
[3]Applied Research, Telcordia Technologies, 445 South Street, Morristown, NJ 07960

ABSTRACT

Ontologies are a central component of the Semantic Web (SW) infrastructure. The design and construction of domain ontologies and taxonomies is a human intensive process which requires allocation of huge resources in terms of cost and time. For the SW to scale and become feasible, approaches that reduce human effort and resource commitments need to be investigated urgently. Towards this end, we present a framework for automated taxonomy construction based on a large corpus of documents, a first step towards large scale, automated ontology construction. Our approach involves: (a) extraction of an appropriate sample from a data set; (b) clustering the documents resulting in a hierarchy; (c) taxonomy extraction from this hierarchy; and (d) assignment of labels to the extracted nodes in the taxonomy. The above framework draws upon a suite of statistical clustering (SC) and natural language processing (NLP) techniques. The variations in each part of the approach are explored in detail and form the basis of an experimentation framework. Metrics are proposed to evaluate the taxonomy generated in comparison to a gold standard, to estimate the impact of these variations. In particular, we perform our experiments in the domain of medical informatics by using the MEDLINE® database as the document collection, and the MeSH taxonomy as the gold standard. Insights, learned from these initial experiments are presented and discussed.

Categories and Subject Descriptors

H.3.1 [**Content Analysis and Indexing]:** Thesauruses, H.3.3 [**Information Search and Retrieval]:** Clustering, I.5.3 [**Clustering]:** Algorithms, Similarity Measures, I.2.6 [**Learning]:** Concept Learning, Knowledge Acquisition, I.2.7 [**Natural Language Parsing]:** Text Analysis

General Terms

Experimentation, Design, Algorithms, Measurement

Keywords

Semantic Web, Ontologies, Automatic Taxonomy Generation, Ontology Learning, K-Means Clustering, Noun Phrase Extraction, MeSH, MEDLINE, Taxonomy Evaluation, Taxonomy Node Labeling

1. INTRODUCTION

The Semantic Web (SW) [1] has been proposed as an extension to the current Web where the content will be machine-understandable. This content is likely to be in the form of documents annotated with metadata descriptions, or data stored in back-end relational databases mapped to structured ontologies (or schemata) describing content in a domain specific manner. Software programs or agents will then be able to gather and analyze information over the web, enabling the development of software to assist humans and streamline business processes both within and across organizational boundaries.

However, machines today understand very little of available web content. In fact, most of the annotations are in the form of tags that describe structure, formatting or presentation information. Approaches for annotation have primarily been manual [2][3], though there have been some attempts at exploring semi-automatic approaches for metadata annotation [4][48]. As observed in these efforts, two resources necessary for realizing the semantic web are: (a) large scale availability of domain specific ontologies; and (b) large scale availability of annotations or metadata descriptions created by using terms, concepts or relationships provided by these ontologies. In this paper, we focus on the former, i.e., addressing the need for domain specific ontologies.

Ontologies are a central component of the SW infrastructure. However, it is well acknowledged that design and construction of ontologies is a labor-intensive process and requires allocation of huge resources in terms of cost and time. For the SW vision to be realized and scale up, it is critical to investigate approaches that reduce human effort and resource commitments. Whereas, the broad goal of the endeavor should be semi-automatic creation of domain ontologies, we begin with an attempt to create an initial thesaurus/taxonomy of concepts using a largely unsupervised learning approach. This taxonomy forms the vital first step in bootstrapping ontologies from textual documents that form an overwhelming proportion of content available on the Web today.

This paper is organized as follows. In Section 2, we review relevant work, focusing on the attempts made by other researchers to address (parts of) this problem. The experimentation framework for taxonomy generation is described in detail in Section 3. The various components of the framework are discussed in detail in Sections 4-10. Section 11 discusses the conclusions and future work.

2. RELATED WORK

Approaches for semi-automatic generation of ontologies or taxonomies from underlying content may be characterized as:

- Supervised machine learning based approaches, which require a large number of training examples, traditionally generated manually.
- NLP approaches applied for generating ontological concepts and relationships. These are based on rules that analyze patterns based on syntactic categories, which requires significant human involvement, making it expensive and infeasible for large scale SW applications.
- SC methods have been used to partition data sets, categorize search results and visualize data. However, they have not focused on generating labels for clusters and creation of new taxonomies.

WWW 2004, May 17-22, 2004, New York, NY USA.
ACM xxx.

Machine learning approaches are for the most part supervised, where a set of manually generated positive and negative training examples are used. An approach using the concept forming system COBWEB [16] has been used to perform incremental conceptual clustering on structured instances of concepts extracted from the web [10]. Experimental and theoretical results on learning the CLASSIC description logic were presented in [32], and were used to construct concept hierarchies. An approach to bootstrap a classification taxonomy based on a set of structured rules was proposed in [35]. A supervised approach presented in [34], supports semi-automatic and incremental bootstrapping of a domain-specific information extraction system.

Empirical and corpus-based NLP methods to build domain specific lexicons have been proposed in [11] and used in [4]. Approaches that learn meanings of unknown words based on other word definitions in the surrounding context have been presented in [12][13]. Case-based methods, that match unknown word contexts against previously seen word contexts are described in [14][15]. Approaches presented in [25][26] apply shallow parsing, tagging and chunking, along with statistical techniques to extract terminologies or enhance existing ontologies. Full parse tree construction followed by decomposition into elementary dependency trees has been used to create medical ontologies from French text corpora in [29]. In [30], a thesaurus is built by performing clustering according to a similarity measure after having retrieved triples from a parsed corpus.

Linguistic structures such as verbs, appositives and nominal modifications have been used to identify hypernymic propositions in the biomedical text [17]. Lexico-syntactic patterns have been investigated for inferring hyponymy from textual data in [7]. Salient words and phrases extracted from the documents are organized hierarchically using subsumption type co-occurrences in [27]. A description of supervised and unsupervised approaches to extract semantic relationships between terms in a text document is presented in [24]. A generalized association rule algorithm proposed in [31] detects non-taxonomic relationships between concepts and also determines the right level of abstraction at which to establish the relationship.

Effectively mining relevant information from a large volume of unstructured documents has received considerable attention in recent years [18][19][20]. A survey on the use of clustering in Information Retrieval is presented in [40]. Document clustering has been used for browsing large document collections in [21], using a "scatter/gather" methodology. These approaches create vector space representations of documents and use Euclidean or cosine distance-based similarity metrics like the Euclidean to extract clusters from groups of documents. Clustering of Web documents to organize search results has been proposed in [22][38]. Physicists have used clustering to find the spatial grouping of stars into galaxies [39]. An approach that pre-processes documents by applying background knowledge in order to improve the clustering results was proposed in [23].

An interesting framework for hybrid approaches, combining the above techniques is presented in [36]. The Thematic Mapping System [8] developed at Verity, Inc. and the lexon mining approach [28] most closely reflects our perspective. A complementary approach that uses the structure and content of HTML-based pages on the Web to generate ontologies is presented in [9]. Hybrid approaches have also been used to automate semantic annotation, a closely related task, examples of which are the SemTag [4] and OntoMate – Annotizer systems [3], and the Semagix content management platform [48].

In view of the above interesting work based on component technologies, we present a comprehensive framework that combines some of these components, and consists of the following novel features:

- An experimental framework combining SC, NLP and other customized techniques for taxonomy generation.
- Exploitation of the statistics generated during the clustering process to extract a more meaningful taxonomy. Identification of statistical parameters that characterize the notion of "differentiation" in the taxonomic structure.
- Techniques for automatic generation and refinement of labels for nodes in the final taxonomy.
- Investigation of the impact of various components of the framework on the quality of the taxonomy generated, based on metrics designed for this purpose.
- Initial validation of our approach using a real world data set, the MEDLINE® database and real world taxonomy, the MeSH thesaurus.

The taxonomy generation framework is discussed next.

3. THE TAXONOMY GENERATION FRAMEWORK

We now identify and discuss the basic components of framework for generating taxonomic/thesauri structures from textual documents (**Figure 1**).

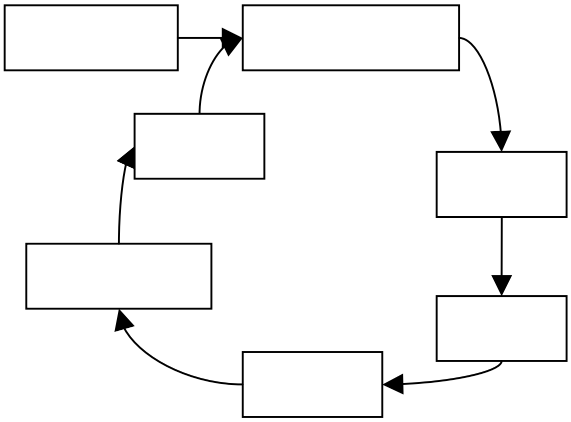

Figure 1: The Taxonomy Generation Framework

Data Extraction and Sampling A *gold standard taxonomy* (the Medical Subject Headings (MeSH) [5]) is chosen and text documents relevant to the gold taxonomy are sampled from the MEDLINE® bibliographic database. We chose the sub-tree under the concept *Cardiovascular Diseases* consisting of 339 concepts. Citations that were annotated by concepts appearing in this sub-tree of the *gold standard taxonomy* and had abstracts associated with them and were chosen. The citations were sampled at different sizes using different techniques based on the underlying distribution of the documents *wrt* the concepts in the taxonomy, for e.g., uniform *vs.* density biased sampling.

NLP techniques for Pre-processing NLP techniques such as part of speech tagging and chunk parsing are used to extract noun phrases from the abstracts. Some variations that can be explored are extracting simple (1-2 words long), macro (2-3 words long) or mega (3-5 words long) noun phrases. Another variation is to choose not to pre-process the documents.

Document Indexing The abstracts (documents) are mapped to a vector space, the dimensions of which could either be words or extracted phrases. Word based indexing may be used in conjunction with noun phrase extraction.

K-Means Clustering Clusters of documents are identified by using a bisecting K-Means strategy, where euclidean or cosine based distance is computed between the document vectors. Interesting variations are related to the cluster quality measures and the type of distance metric (Euclidean *vs.* cosine). Another variation is the use of term vectors to determine term clusters. Document clustering is preferred over term clustering, as in most real data sets there are more terms than documents, giving the SC algorithm a greater discerning power to differentiate clusters.

Taxonomy Extraction The hierarchy generated by the above process is an artifact of the clustering process and does not capture the notion of taxonomy. According to our *taxonomy extraction hypothesis*, ***nodes at lower levels in the taxonomy should capture subject categories that correspond to a narrower information space as compared to nodes at higher levels, and successive levels in the taxonomy should be sufficiently differentiated to be of interest to the user***. The notion of differentiation is captured by the difference in the “cohesiveness” (defined later) between successive layers of the taxonomy. The taxonomy designer suggests a list of cohesiveness levels, based on which the taxonomy extraction algorithm extracts a subset of nodes from the clustering hierarchy and identifies the taxonomic structure. The levels of “cohesiveness” are parameters that can be *tuned* to generate a taxonomy according to the user’s perspective.

Label assignment and smoothing The centroids of the extracted clusters are analyzed to assign a set of potential labels to the nodes in the taxonomy. Terms corresponding to the K highest weighted dimensions in the centroid vector are chosen. Various techniques such as propagation of labels to parent nodes, use of thesauri such as WordNet or the UMLS Metathesaurus can be used refine the labels. Lexico-syntactic patterns [7] can be used to identify potential *subClassOf* relationships.

Taxonomy Evaluation Finally, the generated taxonomy is evaluated *wrt* the gold standard taxonomy using a variety of different measures. These measures capture the content-based similarity (i.e., overlap between the labels extracted) and the structural similarity (i.e., consistency of parent-child relationships) between the two taxonomies.

We now enumerate the dimensions of our experimental framework, based on the variations discussed above.

```
1. Sampling:
   a. Uniform sampling
   b. Density biased sampling
2. Natural Language Processing
   a. No Tagging/Chunking
   b. Noun Phrases: (i) Simple, (ii) Macro, (iii)
      Mega
   c. Verb Phrases
3. Indexing:
   a. Term-based dimensions: Word-based vs. Phrase
      based
   b. SVD eigenvector-based dimensions: Word-based
      vs. Phrase-based
4. Clustering
   a. Document based clustering
   b. Term based clustering
5. Distance Measures
   a. Euclidean
   b. Cosine
6. Cluster Quality Measures:
   a. Internal Measures:
      (i) Pair wise distance,
      (ii) Distance from Centroid
   b. External Measures
7. K-Means Number of Iterations
8. Label assignment:
   a. Threshold: (Value of Top K)
   b. Use of Noun Phrase Matching
   c. Use of Taxonomic Label Propagation
9. Use of Thesauri: Yes/No
10.Use of Lexico-Syntactic Patterns: Yes/No
```

The impact of the some of the dimensions of the above framework on the quality of the taxonomy will be investigated later in the paper. The other dimensions will be investigated in future work. We now discuss the individual components of the Taxonomy Generation Framework in greater detail.

4. SAMPLING THE DATA SET

A subset of the MEDLINE® bibliographic database satisfying the following conditions is extracted: (a) the MEDLINE citation should be annotated by one of the 339 concepts present in the gold taxonomy, i.e. the MeSH sub-tree under the concept *Cardiovascular Diseases*; (b) the concepts that annotate the citation should be identified as “preferred”; and (c) the citation should have a non-empty abstract. We investigate two possible sampling techniques: uniform random sampling and density biased sampling.

“Uniform random sampling is frequently used in practice and also frequently criticized because it will miss small clusters. Many natural phenomena are known to follow Zipf’s distribution and the inability of uniform sampling to find small clusters is of practical concern” [37]. In the context of our approach, sampling is likely to be biased in such a way as to produce a taxonomy containing concepts which appear only in a large number of MEDLINE® citations. Hence, we adopt the approach of density biased sampling as proposed in [37] where we probabilistically under-sample dense regions, i.e., concepts that appear as annotations of a large number of MEDLINE® citations; and over-sample light regions, i.e., concepts that appear as annotations of a small number of MEDLINE® citations. Density biased sampling relies on the *a priori* approximate grouping of data points in the sample. It then samples points from these groups whilst ensuring that dense regions are under-sampled and sparse regions over-sampled. The advantage we have in our experiment is that we know exactly what these groups are *a priori*. This enables us to greatly simplify the sampling process in our experiments. As discussed in [37], the data sets sampled have the following characteristics:

- Given a MeSH concept, documents are selected with a uniform probability. The probability function is:

$$\mathrm{f}(\mathrm{Concept}_i) = \frac{\alpha}{\sqrt{\mathrm{size}(\mathrm{Concept}_i)}}$$

- The sample is density preserving and biased by group size.
    - For a given sample size M, the value of α is given by:

$$\alpha = \frac{\mathrm{M}}{\sum_{i=1}^{339} \sqrt{\mathrm{size}(\mathrm{Concept}_i)}}$$

5. NATURAL LANGUAGE PROCESSING

The PhraseX program developed at the National Library of Medicine is used to extract Noun Phrases from the documents. PhraseX extracts noun phrases from text by referring to the syntactic structure provided by the SPECIALIST minimal commitment parser. The SPECIALIST minimal commitment parser relies on the SPECIALIST Lexicon as well as the Xerox stochastic tagger [41]. The output contains simple noun phrases. The authors in [42] refer to these phrases as "core noun phrase," that is, a noun phrase with no modification to the right of the head.

The SPECIALIST parser is based on the notion of barrier words [43] which indicate boundaries between phrases. After lexical look-up and resolution of category label ambiguity by the tagger, complementizers, conjunctions, modals, prepositions, and verbs are marked as boundaries. Subsequently, boundaries are considered to open a new phrase (and close the preceding phrase). Any phrase containing a noun is considered to be a (simple) noun phrase, and in such a phrase, the right-most noun is labeled as the head; all other items (other than determiners) are labeled as modifiers. An example of the output from the SPECIALIST parser is given in (2) for the input in (1).

```
(1)  Kupffer  cells  from  halothane-exposed  guinea
     pigs    carry    trifluoroacetylated    protein
     adducts.
```

```
(2)[[mod([lexmatch(['Kupffer']),
          inputmatch(['Kupffer']),tag(noun)]),
     head([lexmatch([cells]),
     inputmatch([cells]),tag(noun)])],
    [prep([lexmatch([from]),
           inputmatch([from]),tag(prep)]),
     mod([lexmatch([halothane]),
          inputmatch([halothane]),tag(noun)],
          punc([inputmatch([-])]),
          mod([lexmatch([exposed]),
               inputmatch([exposed]),tag(adj)]),
          head([lexmatch(['guinea pigs']),
                inputmatch([guinea,pigs]),
                              tag(noun)])],

    [verb([lexmatch([carry]),inputmatch([carry]),
                                     tag(verb)])],
    [mod([lexmatch([trifluoroacetylated]),
          inputmatch([trifluoroacetylated]),
                                  tag(adj)]),
     mod([lexmatch([protein]),
          inputmatch([protein]),tag(noun)]),
     head([lexmatch([adducts]),
           inputmatch([adducts]),tag(noun)]),
     punc([inputmatch(['.'])])]]
```

The underspecified structure produced by the SPECIALIST parser serves as the basis for the extraction of noun phrase strings by PhraseX. In addition to the simple noun phrase (labeled as "simp" in output), PhraseX identifies two additional structures. One of these is the complex noun phrase in which a head is followed by contiguous prepositional phrases to its right ("macro"). The first preposition in this structure can be anything, but all the rest must be "of". The second structure is not a canonical syntactic phenomenon, but may be important for information processing. Such a phrase includes all the content words that occur in a sentence either to the left or the right of a finite verb ("mega"). Examples of these strings as extracted from the syntactic structure in (2) are given in (3).

```
(3) 00000000|simp|kupffer cells
    00000000|simp|halothane exposed guinea pigs
    00000000|simp|trifluoroacetylated protein
                                         adducts
    00000000|macro|kupffer cells from halothane
                              exposed guinea pigs
    00000000|mega|kupffer cells from halothane
                              exposed guinea pigs
    00000000|mega|trifluoroacetylated protein
                                         adducts
```

6. DOCUMENT INDEXING

There are two possibilities related to indexing the documents:

- Terms are used as dimensions of the underlying vector space as in the SMART Indexing and Retrieval Engine [44].
- The Latent Semantic Indexing approach [47], where a *Singular Value Decomposition (SVD)* analysis identifies the underlying eigenvectors. These are used as dimensions of a common “latent” space in which both term and document vectors can be represented.

Either technology can be used with either words or phrases as features. The documents can be pre-processed to extract noun phrases, which can then be indexed by using either of the above approaches. Alternatively, the raw text bag of words, after removal of stop words, can be indexed.

7. CLUSTERING THE DATA SET

The document vectors generated by the document indexing engine undergo a clustering process, using a bisecting k-means algorithm. A hierarchical cluster tree is generated. Consider a set of document vectors $\boldsymbol{D} = \{d_1, ..., d_M\}$ in the Euclidean space $\mathbf{R}^N$. Let the centroid of the set be denoted by:

$$\mathrm{m}(\boldsymbol{D}) = \frac{1}{\mathrm{M}}\sum_{i=1}^{\mathrm{M}} \mathrm{d}_i$$

The cohesiveness of the set (also known as intra-cluster cohesiveness) is defined as:

$$\mathrm{c}(\boldsymbol{D}) = \frac{1}{\mathrm{M}}\sum_{i=1}^{\mathrm{M}} \cos(\mathrm{d}_i, \mathrm{m}(\boldsymbol{D}))$$

Let $\{\pi_i\}^k_{i=1}$ be a partition of $\boldsymbol{D}$ with the corresponding centroids $m_1 = m(\pi_1), ..., m_k = m(\pi_k)$. The quality of the partition increases if the intra-cluster cohesiveness increases. Thus the quality $\boldsymbol{Q}$ of the partition $\{\pi_i\}^k_{i=1}$ is given by:

$$\boldsymbol{Q}(\{\pi_i\}_{i=1}^{k}) = \frac{1}{k}\sum_{i=1}^{k} \mathrm{c}(\pi_i)$$

We start with the set of all the documents as the initial cluster. Let $C_1, ..., C_i$ be the set of clusters at i^{th} iteration. We choose a cluster S using a selection rule and apply k-means clustering with $k=2$ to give $(i+1)$ clusters. Typically a cluster with the lowest intra-cluster cohesiveness or the one with maximum intra-cluster variance is chosen. We check to determine if there is significant improvement in the partition quality. In case there is, we run k-means on all the $(i+1)$ clusters to stabilize the clusters at this level. Changes in the clusters are noted and the above process is repeated until a significant increase in the quality measure is not seen. The algorithm pseudo-code is presented below.

```
1.  Start with a single cluster $\boldsymbol{D}$ at level = 1.
2.  At tree level = L,
    a.  Select a cluster $\pi_{j,L}$ from the partition
        $\{\pi_{i,L}\}^k_{i=1}$ which has the lowest value for
        $c(\pi_{j,L})$
    b.  Run k-means clustering on $\{\pi_{j,L}\}$ with k = 2
        to obtain a new partition with k+1
        clusters $\{\pi_{i,L+1}\}^{k+1}_{i=1}$. This includes the
        clusters $\{\pi_{j,L+1}, \pi_{k+1,L+1}\}$ generated from
        cluster $\pi_{j,L}$.
3.  Check if $\boldsymbol{Q}(\{\pi_{i,L+1}\}^{k+1}_{i=1})$ is significantly
    greater than $\boldsymbol{Q}(\{\pi_{i,L}\}^{k}_{i=1})$
4.  If there are significant gains,
    a.  Copy the centroids to initialize a new
        partition at level L+1, i.e., $m_i = m(\pi_{i,L+1})$
    b.  Establish the following relationships:
        i.   $child(\pi_{j,L}) = \pi_{j,L+1}$
       ii.   $child(\pi_{j,L}) = \pi_{k+1,L+1}$
      iii.   $child(\pi_{i,L}) = \pi_{i,L+1}$ for other clusters.
    c.  Run k+1 means clustering on $\{\pi_{i,L+1}\}^{k+1}_{i=1}$ to
        stabilize the clusters at level L+1
    d.  Goto step 2.
5.  Stop.
```

It should be noted that the hierarchical cluster tree is an artifact of the clustering algorithm and is **not** the taxonomy that will be generated. As a part of the clustering process, we compute certain parameters that will be useful in extracting the final taxonomy. The parameters are:

- The intra-cluster cohesiveness $c(\pi_i)$. This determines the differentiation in meaning between successive levels of the extracted taxonomy.
- The centroid vector $m(\pi_i)$. This is used to generate potential labels corresponding to a cluster.
- The parent child relationships between the clusters generated at the various levels.

8. TAXONOMY EXTRACTION

According to our taxonomy extraction hypothesis, *nodes at lower levels in the taxonomy should capture subject categories that correspond to a narrower information space as compared to nodes at higher levels, and successive levels in the taxonomy should be sufficiently differentiated to be of interest to the user.* The notion of differentiation is captured by the difference in the ***cluster cohesiveness*** between successive layers of the hierarchical cluster tree. The taxonomy creator or user is expected to suggest a set of cohesiveness levels which correspond to differentiation between the various layers of the taxonomy. In the course of our experimentation, it was observed that the successive values of cohesiveness down a cluster hierarchy are ***monotonically increasing*** in value. In general, this will be an iterative process involving display of the raw clustering and labeling results to the user. This will give him/her a better idea of how to set up the cohesiveness levels to produce the desired taxonomy. The levels of cohesiveness are thus parameters which can be varied to better "tune" a taxonomy that corresponds to the creator's perspective of the information domain. The process of interaction between the taxonomy creator and the TaxaMiner system and "tuning" of the parameters are beyond the scope of this paper and will be addressed in our future work.

Given a set of cohesiveness parameters, the taxonomy extraction algorithm extracts a subset of nodes from the clustering hierarchy and identifies the taxonomic structure (**Figure 2**). The input to this algorithm is a cluster hierarchy (H) with the computed cohesiveness measure $c(\pi_i)$ and a set of thresholds: $\mu_1 \geq \ldots \geq \mu_N$ and the output is an extracted taxonomy (T).

A set of paths belonging to a tree T is denoted by $paths(\boldsymbol{T}) = \{p_1, \ldots, p_M\}$ and contains the paths originating from the root of the tree and ending at the leaf nodes of the tree. The paths corresponding to the hierarchical cluster ***H*** in **Figure 2** are:
*paths(**H**) = {"DSSS…", "DHH$_4$ H$_4$ H$_4$…", "DHKLH$_3$…", …}.*

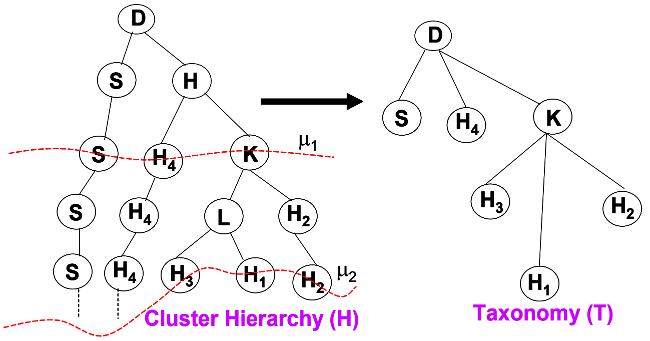

Figure 2: Taxonomy Extraction from Hierarchical Cluster Tree

Each node in ***H*** corresponds to a cluster of documents. A set of selected nodes corresponding to a cohesiveness threshold μ_j is denoted by *selectedNodes(μ_j)* and identifies clusters π_j s.t. $c(\pi_j)$ is closest to μ_j. The selected nodes as illustrated in **Figure 2** are:

selectedNodes(μ_1) = {S, H$_4$, K}
selectedNodes(μ_2) = { H$_3$, H$_1$, H$_2$}

We now present an algorithm for taxonomy extraction.

```
1.  For each path $p_i$ in paths($\boldsymbol{H}$) do
    a.  For j = 1 to N do
        i.   Find nodes A and B in $p_i$ s.t. $c(A) \leq \mu_j$
             $\leq c(B)$
       ii.   If $(\mu_j - c(A)) \leq (c(B) - \mu_j)$
             Insert A in selectedNodes($\mu_j$)
             Else, Insert B in selectedNodes($\mu_j$)
2.  Collapse $\boldsymbol{H}$: For i = 1 to N do
    a.  For each Node A in selectedNodes($\mu_i$) do
        i.   If i>1,
             Find ancestor(A) in selectedNodes($\mu_{i-1}$)
       ii.   If i=1, ancestor(A) = root($\boldsymbol{H}$)
      iii.   Delete all nodes from on the path from
             A to ancestor(A)
       iv.   Establish ancestor(A) as the parent of
             A in the extracted taxonomy $\boldsymbol{T}$
3.  End Extract Taxonomy
```

9. TAXONOMY LABELING

Once the relevant taxonomy nodes have been extracted from the cluster hierarchy tree, the following steps are performed:

- For each node in the extracted taxonomy, a set of potential labels that are extracted.
- These sets of labels are then pruned and smoothened using both the noun phrases extracted from the dataset and taxonomic label propagation.

9.1 Label Extraction and Assignment

Label extraction and assignment depends on the underlying indexing technique used to create the vector space. The extraction of the top K terms that contribute most to the centroid vector can be implemented in the following two ways:

- In the case of SMART [44], terms and documents have their own underlying vector spaces. Hence, we simply choose the top K values of the centroid vector and determine the terms which contribute to the top K terms.
- In the case of the LSI [47], terms and documents are represented in the same "latent" space. This enables us to compute the (Euclidean or cosine) distance between the centroid vector and the term vectors.

Given a cluster node π_i, we define the *labels(π_i)* to contain the labels assigned to the cluster in the taxonomy tree.

$$\text{childLabels}(\pi_i) = \bigcup_{A \in \text{children}(\pi_i)} \text{labels}(A)$$

$$\text{parentLabels}(\pi_i) = \text{labels}(\text{parent}(\pi_i))$$

$$\text{taxonomyLabels}(\boldsymbol{T}) = \bigcup_{A \in \boldsymbol{T}} \text{labels}(A)$$

9.2 Label Smoothing and Propagation

Having assigned labels to each of the nodes in the extracted taxonomy, the first challenge is to determine which of the K labels are relevant to the node and which are spurious. Also, the same labels can appear in multiple nodes of the taxonomy. Two approaches that can be used for label smoothing are:

1. **Use of Noun Phrase information:** The noun phrases extracted from the documents are stored in a lexicon. Individual words are combined into potential phrases in the lexicon, thus reducing the number of labels. Let the lexicon be denoted by *lexicon(T)*
2. **Use of Taxonomic Label Propagation:** The assignment of labels across multiple nodes in the taxonomic structure is used to propagate labels across different levels of the taxonomy and thereby reduce the number of spurious labels.

In this paper, we focus only on taxonomic label propagation. Some heuristics for label propagation are:

- **Propagate to Child:** If a label appears both in the parent and one or few children, the label will be propagated to the child and removed from the parent. A parent node in a taxonomy is a generalization of its children. Hence the parent should not have a label that only one or few of its children have.
- **Propagate to Parent:** If a label has been assigned to all the children of a node, the label will be propagated to the parent and removed from all the children nodes at which it appears. If every child of a node in a taxonomy has a label that the node itself has, having that label in the parent node suffices to convey the fact that children of this node also talk about the concept that the label represents.

The algorithm for label propagation and smoothing is as follows.

```
1.  Start with the Root(T)
2.  For each cluster node $\pi_i$ at level L do
    a.  For cluster node $\pi_j \in$ children($\pi_i$) do
        i.  If $\Delta$ = labels($\pi_i$) $\cap$ labels($\pi_j$) $\neq \phi$
       ii.  labels($\pi_i$) = labels($\pi_i$) - $\Delta$
3.  End Propagate to Children
4.  Start with cluster nodes in leaves(T)
5.  For each cluster node $\pi_i$ at level L do
    a.  If $\Delta$ = labels($\pi_i$) $\cap$ childLabels($\pi_i$) $\neq \phi$
    b.  labels($\pi_i$) = labels($\pi_i$) + $\Delta$
    c.  For $\pi_j \in$ children($\pi_i$) do
        i.  labels($\pi_j$) = labels($\pi_j$) - $\Delta$
6.  End Propagate to Parent
7.  End Label Propagation
```

10. EXPERIMENTAL EVALUATION

We now discuss metrics used to evaluate the quality of the taxonomy generated by our algorithms. Experiments that investigate the impact of the various factors enumerated in the taxonomy generation framework (Section 3) on the quality of the taxonomy generated are discussed.

10.1 Taxonomy Quality Metrics

Metrics have been developed for approximate tree matching using edit distance in [45]. Whereas we plan to develop more sophisticated and sensitive metrics for taxonomy quality based on the ideas in [45], in our current work, we propose simple and pragmatic metrics to evaluate the generated taxonomy.

1. **Content Quality:** This component measures the overlap in the labels present in the generated Taxonomy, $\boldsymbol{T_{gen}}$ and the gold standard taxonomy $\boldsymbol{T_{gold}}$.
2. **Structural Quality:** This component measures the structural validity of the labels, i.e., when two labels appear in a parent child relationship in $\boldsymbol{T_{gold}}$, they should appear in a consistent relationship (parent-child or ancestor-descendant) in $\boldsymbol{T_{gen}}$.

The algorithm to compute the quality metrics is presented below.

```
1.  contentQ = 0
2.  structQ = 0
3.  For each $\pi_i \in T_{gen}$ do
    a.  matchLabels($\pi_i$) = $\phi$
    b.  For each label$_j$ $\in$ labels($\pi_i$) do
        i.  If  label$_j$  $\in$  taxonomyLabels($T_{gold}$)  and
            label$_j$ $\notin$ matchLabels($\pi_K$), 1 $\leq$ K $\leq$ i-1
              contentQ = contentQ + $\delta$
              add label$_j$ to matchLabels($\pi_i$)
       ii.  NumComparisons = NumComparisons + 1
4.  Normalize: contentQ = contentQ/NumComparisons
5.  End Content Quality Computation
6.  Start with Root($T_{gen}$)
7.  For each cluster node $\pi_i$ at level L do
    a.  For each $\pi_j \in$ children($\pi_i$) do
    b.  LabelPairSet = matchLabels($\pi_i$) ×
                                         matchLabels($\pi_j$)
        i.  For each <pLabel,cLabel>
                                        $\in$ LabelPairSet do
       ii.  If pLabel = parent(cLabel) or pLabel =
            ancestor(cLabel) in $T_{gold}$
                    structQ = structQ + 1
                    Exit ForLoop (Begins at 7(a))
      iii.  NumComparisons = NumComparisons + 1
8.  Normalize: structQ = structQ/NumComparisons
9.  End Structural Quality Computation
```

10.2 Experimental Results

We present an initial set of experiments evaluating the impact of the following on the (content and structural) quality of the taxonomies generated:

- The effect of varying the size of the data sets.
- The effect of varying the number of labels extracted.
- The effect of pre-processing the document set using limited NLP techniques (Noun Phrase Extraction).

The content and structural quality measures defined in the previous section will be used with the following caveats:

- In the current set of experiments, we have generated only 50 levels of the clustering hierarchy. We plan to generate more levels in further experiments which will lead to better results.
- A subject matter expert is required for setting the threshold levels for taxonomy extraction, i.e., the μ values discussed in Section 8. In our current experiments we have assigned μ values automatically based on the minimum and maximum values of cohesiveness. We believe that the involvement of an expert would significantly improve the quality measures.

The gold standard taxonomy and a sample learned taxonomy are illustrated in the **Appendix** (**Figure 7** and **8**) at the end of this paper. We begin with a set of experiments involving multiple data sets that have been pre-processed using NLP techniques (**Figure**

3). Taxonomy content quality measures are computed for each of the taxonomies for different values of *K* (the size of the label sets extracted at each cluster node in the taxonomy).

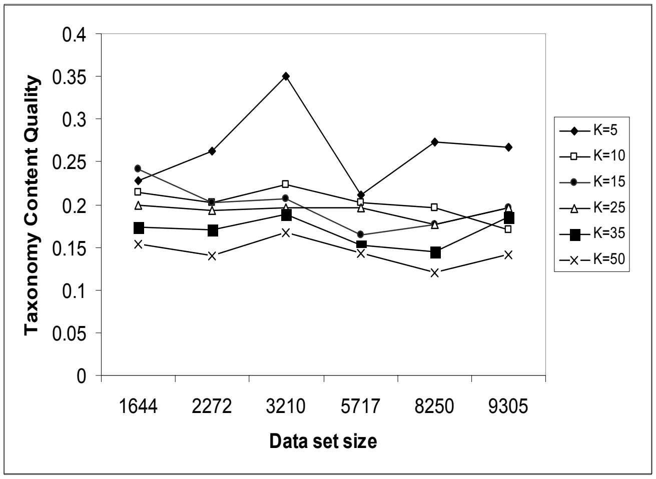

Figure 3: Taxonomy Content Quality for different sizes of the data sets, and extracted label sets/cluster node

Some interesting trends observed in **Figure 3** above are:

- Increasing the data set size does not necessarily increase the content quality of the taxonomy generated. In fact, we notice a trend that suggests that the taxonomy quality peaks and then tends to deteriorate for larger data sets.
- Extracting a lesser number of labels for each cluster node (the value of *K*) gives better results for the content quality of the generated taxonomy.
- A few "crossings" are observed as the value of *K* increases; i.e., higher values of *K* outperform the algorithm for lower values for some data points.

In the next figure, we repeat the same set of experiments as in the case of **Figure 3**, but evaluate the structural quality of the generated taxonomies.

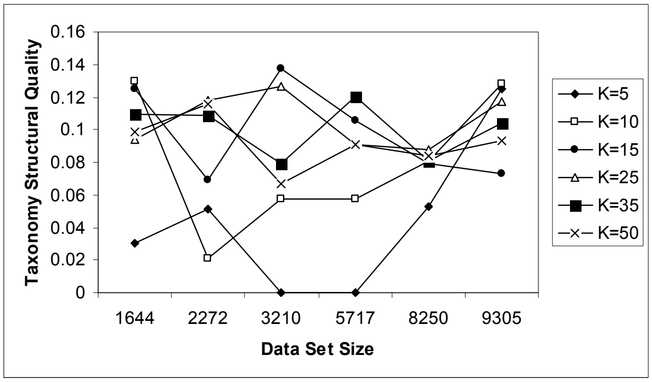

Figure 4: Taxonomy Structure Quality for different sizes of the data sets and extracted label sets/node

Some interesting trends observed in **Figure 4** above are:

- Compared to the content quality measure, the structural quality measure has a lower trend of values.
- In most cases there seems to be a beneficial impact of the increase in the data set size on, the structural quality measure though this aspect needs to be investigated further.

In the next set of experiments, we investigate the impact of pre-processing the document set using limited NLP techniques, such as noun phrase extraction. For a particular value of the number of labels extracted/node, *K=35*, the content (**Figure 5**) and structural (**Figure 6**) quality of the generated taxonomies are evaluated.

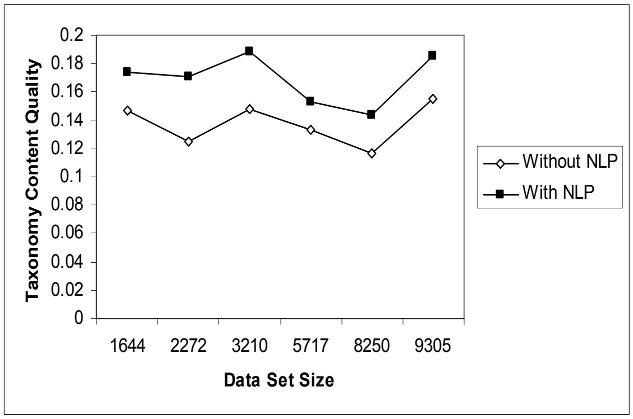

Figure 5: Comparison of Taxonomy Content Quality with and without NLP-based pre-processing (*K=35*)

The content quality graphs observed in the above figure mirror each other in both the cases (with and without NLP pre-processing) and suggest that there is definite value in using limited NLP techniques to pre-process a document set.

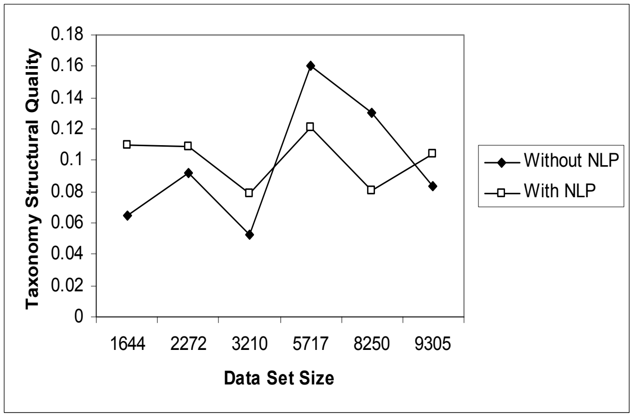

Figure 6: Comparison of Taxonomy Structure Quality with and without NLP-based pre-processing (*K=35*)

The structural quality graphs observed in the above figure do not show a consistent trend or pattern. However, there are quite a few graph crossings which indicate that approaches with and without NLP out-perform each other at various data points. The interested reader may visit the project website, http://cgsb2.nlm.nih.gov/~kashyap/projects/TaxaMiner for more examples of taxonomies generated using our techniques.

10.3 Discussions and Insights

The experiments discussed above are a component of extensive ongoing work in evaluating a suite of taxonomy generation techniques. They have provided us with some interesting insights, which indicate further areas of research and investigation.

The fact that taxonomy content quality decreases after an observed peak, suggests that there might be an optimal data set size for good quality taxonomy extraction. Also increasing the data set size may not add new information content and probably introduces noise into the generated taxonomy. An interesting research problem would be to come up with decision procedures to estimate the optimal size of a data set to generate an optimal taxonomy, especially in the absence of a gold standard.

Increasing the number of labels extracted for each cluster nodes (value of *K*), increases the content quality of the generated taxonomy a little bit, but then for larger values of *K*, the content quality deteriorates significantly. For really high values of *K*, we see almost a straight line, which even overwhelms the impact of increasing the size of the data set. This is probably due to a large number of spurious labels being generated, indicating that there is probably an optimum value of the label set that can be generated, another area of potential investigation.

The lower values of structural quality (as opposed to content quality) suggest that a deeper investigation is needed to obtain better results. We expect meaningful user input in the form of judiciously chosen cohesiveness thresholds (μ values) to alleviate the problem by identifying the correct level of differentiation and alignment. In our current experiments, we have implemented heuristics that identify these μ values automatically. These heuristics can be further enhanced, in conjunction with reference taxonomies of the domain to automatically recommend a range of μ values to the user for taxonomy extraction.

The structural quality seems to improve at higher values of the data set size, especially when the value of *K* is lower. This suggests that there might be an optimal combination of the number of documents in the data set and the number of labels extracted that might give good results.

For certain applications like information filtering and semantic annotation, the content quality measure might have more importance as opposed to the structural quality measure. We need to investigate a composite measure that gives different weights to the content and structural components and configure the taxonomy generation algorithm appropriately.

Pre-processing the documents using NLP techniques gives better taxonomy content quality. However, we observe that NLP and non-NLP approaches out-perform each other at different points. A deeper investigation into this phenomenon would enable us to develop hybrid SC and NLP approaches to optimize a combination of content and structural taxonomy quality.

11. CONCLUSIONS AND FUTURE WORK

The main contribution of this paper is a comprehensive approach and framework for the difficult, and yet important problem for bootstrapping taxonomies from textual data. In contrast to other approaches, that address components of the problem, we present a comprehensive process and strategy that minimizes the involvement of a domain expert in creating a taxonomy. Some of the novel features of our work are:

- A systematic experimental framework that combines and evaluates statistical clustering, NLP and other techniques for taxonomy generation. Design of taxonomy quality metrics and their use to evaluate the impact of the above techniques on the quality of the results generated.
- Exploitation of the statistics generated during the clustering process to extract a more meaningful taxonomy. Identification of statistical parameters that characterize the notion of "differentiation" in the taxonomic structure.
- Techniques for automatic generation and refinement of labels for creating the final taxonomy.
- Initial validation of our approach using a real world data set, the MEDLINE® database and real world taxonomy, the MeSH thesaurus.

Initial experimentation points out interesting insights. One insight suggests that a generated taxonomy consists of intrinsic information content, and analyzing larger data sets and extracting more labels will not necessarily guarantee good results. Human involvement, though minimized is crucial to the process of creating good quality taxonomies. Also, the notion of quality of a taxonomy is a combination of content-based and structure-based components and needs to be specified in an application and domain specific manner. Finally, an optimal strategy for taxonomy generation based on a user configured quality metric involves a joint optimization of various parameters.

This work is an ongoing collaboration between researchers at the National Library of Medicine, LSDIS Lab at the University of Georgia and Applied Research Labs at Telcordia Technologies. Some issues that we are investigating are:

- Algorithmic techniques for improving the structural quality of the generated taxonomies.
- Understand and leverage the human expert, especially in the context of identifying the levels of differentiation in the taxonomy that corresponds to his/her perspective of the application or domain. Combined quality metrics that better reflect the needs of the user.
- Investigation of the notion of an optimal set of parameters for generating a taxonomy. For example, processing a bigger data set can be avoided if we know that the resulting improvement in the taxonomy quality will be negligible.
- Investigation of NLP and other techniques [7] to further refine the taxonomies generated into richer ontologies.

We believe that pragmatic issues as enumerated above are crucial for generating ontologies/taxonomies in a scalable and feasible manner and that we have taken a very important first step in this direction.

12. REFERENCES

[1] T. Berners Lee, J. Hendler and O. Lassila. The Semantic Web. *Scientific American*, May 2001.

[2] J. Kahan, M-R. Koivunen, E. Prud'Hommeaux and R. Swick. Annotea: An open RDF Infrastructure for shared annotations. *Proceedings of the 10th International WWW Conference (WWW 2002)*, Hong Kong, May 2001

[3] S. Handschuh, S. Staab and R. Volz. On Deep Annotation. *Proceedings of the 12th International WWW Conference (WWW 2003)*, Budapest, Hungary. May 2003.

[4] S. Dill et. al. SemTag and SemSeeker: Bootstrapping the Semantic Web via automated semantic annotation. *Proceedings of the 12th International WWW Conference (WWW 2003)*, Budapest, Hungary, May 2003.

[5] MeSH. Medical Subject Headings. Bethesda (MD): National Library of Medicine, 2003. http://www.nlm.nih.gov/mesh/meshhome.html

[6] G Salton, Editor. The SMART Retrieval System – Experiments in Automatic Document Retrieval. Prentice Hall Inc., Englewood Cliffs, NJ 1971.

[7] Hearst, M.: *Automatic acquisition of hyponyms from large text corpora*. In Proceedings of the 14th International Conference on Computational Linguistics. Nantes, France, 1992.

[8] C Y Chung, R. Lieu, J. Liu, A. Luk, J. Mao and P. Raghavan. Thematic Mapping – From Unstructured Documents to Taxonomies. *Proceedings of the 11th International Conference on Information and Knowledge Management (CIKM 2002),* McLean, VA, November 2002.

[9] H. Davulcu, S. Vadrevu and S. Nagarajan. OntoMiner: Bootstrapping and Populating Ontologies from Domain Specific Websites. *Proceedings of the First International Workshop on Semantic Web and Databases (SWDB 2003)*, Berlin, September 2003.

[10] P. Clerkin, P. Cunningham and C. Hayes. Ontology Discovery for the Semantic Web using Hierarchical *Clustering. Proceedings of the Semantic Web Mining Workshop co-located with ECML/PKDD 2001*, Freiburg, Germany, September 2001

[11] E. Riloff and J. Shepherd. A corpus-based approach for building semantic lexicons. In *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing (EMNLP-97)*, Providence, RI, 1997

[12] P. Jacobs and U. Zernik. Acquiring Lexical Knowledge from Text: A Case Study. In *Proceedings of the Seventh National Conference on Artificial Intelligence*, 1988.

[13] P. Hastings and S. Lytinen. The Ups and Downs of Lexical Acquisition. In *Proceedings of the Twelfth National Conference on Artificial Intelligence*, 1994.

[14] R. C. Berwick. Learning Word Meanings from Examples. In *Semantic Structures: Advances in Natural Language Processing*. Lawrence Erlbaum Associates, 1989.

[15] C. Cardie. A Case-based Approach to Knowledge Acquisition for Domain Specific Sentence Analysis. In *Proceedings of the Eleventh National Conference on Artificial Intelligence*, 1993.

[16] D. H. Fisher. Knowledge Acquisition via incremental conceptual clustering. *Machine Learning 2:139-172*, 1987

[17] M. Fiszman, T. C. Rindflesch and H. Kilicoglu. Integrating a Hypernymic Preposition Interpreter into a Semantic Processor for Biomedical Texts. In *Proceedings of the AMIA Annual Symposium on Medical Informatics*, 2003.

[18] B. S. Everitt, S. Landau and M. Leese. Cluster Analysis. Edward Arnold. 4th Edition, May 2001.

[19] Y. Zhang and G. Karypis. Criterion functions for Document Clustering. *Technical Report, U. Minnesota, Dept. of Computer Science, #TR-01-40.*

[20] S. Chakrabarti. Data Mining for Hypertext: A Tutorial Survey. *ACM SIGKDD Explorations*, 1(2):1-11, 2000.

[21] D. R. Cutting, D. R. Karger, J. O. Pedersen and J. W. Tukey. Scatter/Gather: A cluster-based approach to browsing large document collections. In *Annual International Conference on Research and Development on Information Retrieval*, Denmark, 1992.

[22] O. Zamir and O. Etzioni. Web Document Clustering: A Feasibility Demonstration. In *Proceedings of ACM SIGIR Conference*, 1998.

[23] A. Hotho, S. Staab and A. Maedche. Ontology-based Text Clustering. In *Proceedings of the IJCAI 2001 Workshop on Text Learning: Beyond Supervision*, Seattle, USA, 2001.

[24] M. Finkelstein-Landau and E. Morin. Extracting Semantic Relationships between Terms: Supervised vs Unsupervised Methods. In *Proceedings of International Workshop on Ontological Engineering on the Global Information Infrastructure*, Dagstuhl Castle, Germany, May 1999.

[25] B. Daille. Study and implementation of combined techniques for automatic extraction of terminology. In P. Resnick and J. Klavans, editors, *The Balancing Act: Combining Symbolic and Statistical Approaches to Language*, MIT Press, Cambridge, MA, 1996

[26] M. Missikoff, P. Velardi and P. Fabriani. Text Mining Techniques to automatically enrich a Domain Ontology. *Applied Intelligence* 18, 323-340, 2003.

[27] M. Sanderson and B. Croft. Deriving Concept Hierarchies from Text. *International Conference on Research and Development in Information Retrieval (SIGIR 1999)*, 1999.

[28] M. Reinberger, P. Spyns, W. Daelemans and R. Meersman. Mining for Lexons: Applying unsupervised learning methods to create ontology bases.

[29] A. Nazarenko, P. Zweigenbaum, J. Bouaud and B. Habert. Corpus-based identification and refinement of semantic classes. In *Proceedings of the AMIA Annual Symposium*, 1997.

[30] D. Lin. Automatic retrieval and clustering of similar words. In *Proceedings of COLING-ACL-98*, 1998.

[31] A. Maedche and S. Staab. Discovering conceptual relations from text. Technical *Report 399, Institute AIFB, Karlsruhe University*, 2000.

[32] W. W. Cohen and H. Hirsh. Learning the CLASSIC Description Logic: Theoretical and Experimental Results. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Fourth International Conference*, 1994.

[33] A. Borgida and P. F. Patel-Schneider. A semantics and complete algorithm for subsumption in the CLASSIC description logic. *AT&T Technical Memorandum*, 1992.

[34] A. Maedche, G. Neumann and S. Staab, Bootstrapping an Ontology Based Information Extraction System. *Studies in Fuzziness and Soft Computing, INTELLIGENT EXPLORATION OF THE WEB*, P.S. Szczepaniak, J. Segovia, J. Kacprzyk, L.A. Zadeh, Springer, 2003.

[35] H. Suryanto and P. Compton: Learning Classification taxonomies from a classification knowledge based system**.** In *Proceedings of Workshop on Ontology Learning at ECAI-2000*, 2000.

[36] A. Maedche and S. Staab. Ontology learning for the Semantic Web. *IEEE Intelligent Systems*, 16, 2001.

[37] C. R. Palmer and C. Faloutsos. Density Biased Sampling: An Improved Method for Data Mining and Clustering. In Proceedings of ACM SIGMOD International Conference on Management of Data, May 2000

[38] C. Buckley, M. Mitra, J. Walz and C. Cardie. Using clustering and superconcepts within SMART: TREC 6. In *Sixth Test Retrieval Conference (TREC-6)*, Gaithersburg, MD, November 1997.

[39] J. Kepner, X. Fan, N. Buhcall, J. Gunn, R. Lupton and G. Xu. An Automated Cluster Finder: The Adaptive Matched Filter. *The Astrophysics Journal*, 517, 1999.

[40] E. Rasmussen. Clustering Algorithms. In W. B. Frakes and R. Baeza-Yates, editors, *Information Retrieval: Data Structures and Algorithms*, Prentice Hall, 1992

[41] D. R. Cutting, J. Kupiec, J. O. Pedersen and P. Sibun. A practical part-of-speect tagger. In *Proceedings of the Third Conference on Applied Natural Language Processing*, 1992.

[42] R. Weischedel, M. Meteer, R. Schwartz, L. Ramshaw and J. Palmucci. Coping with ambiguity and unknown words through probabilistic models. *Computational Linguistics*, 19(2), 1993.

[43] K. W. F. Tersmette, A. F. Scott, G. W. Moore and R. E. Miller. Barrier word method for detecting molecular biology multiple word terms. *Proceedings of the 12th Annual Symposium on Computer Applications in Medical Care*, 1988

[44] G. Salton. The SMART Retrieval System – Experiments in Automatic Document Retrieval, Prentice Hall, 1971.

[45] J. T. Wang, K. Zhang, K. Jeong and D. Shasha. A System for Approximate Tree Matching. *IEEE Transactions on Knowledge and Data Engineering*, 6(4), August 1994.

[46] Y. Park, R. Byrd and B. Boguraev. Towards Ontology on Demand. *Proceedings of the ISWC Workshop on Semantic Web Technologies for Searching and Retrieving Scientific Data*, October 2003.

[47] Deerwester, S., Dumais, S. T., Landauer, T. K., Furnas, G. W. and Harshman, R. A., "ndexing by latent semantic analysis. *Journal of the Society for Information Science, 41(6)*, 391-407, 1990.

[48] B. Hammond, A. Sheth, and K. Kochut. Semantic Enhancement Engine: A Modular Document Enhancement Platform for Semantic Applications over Heterogeneous Content. In *Real World Semantic Web Applications*, V. Kashyap and L. Shklar, Eds., IOS Press,

APPENDIX: TAXONOMIES

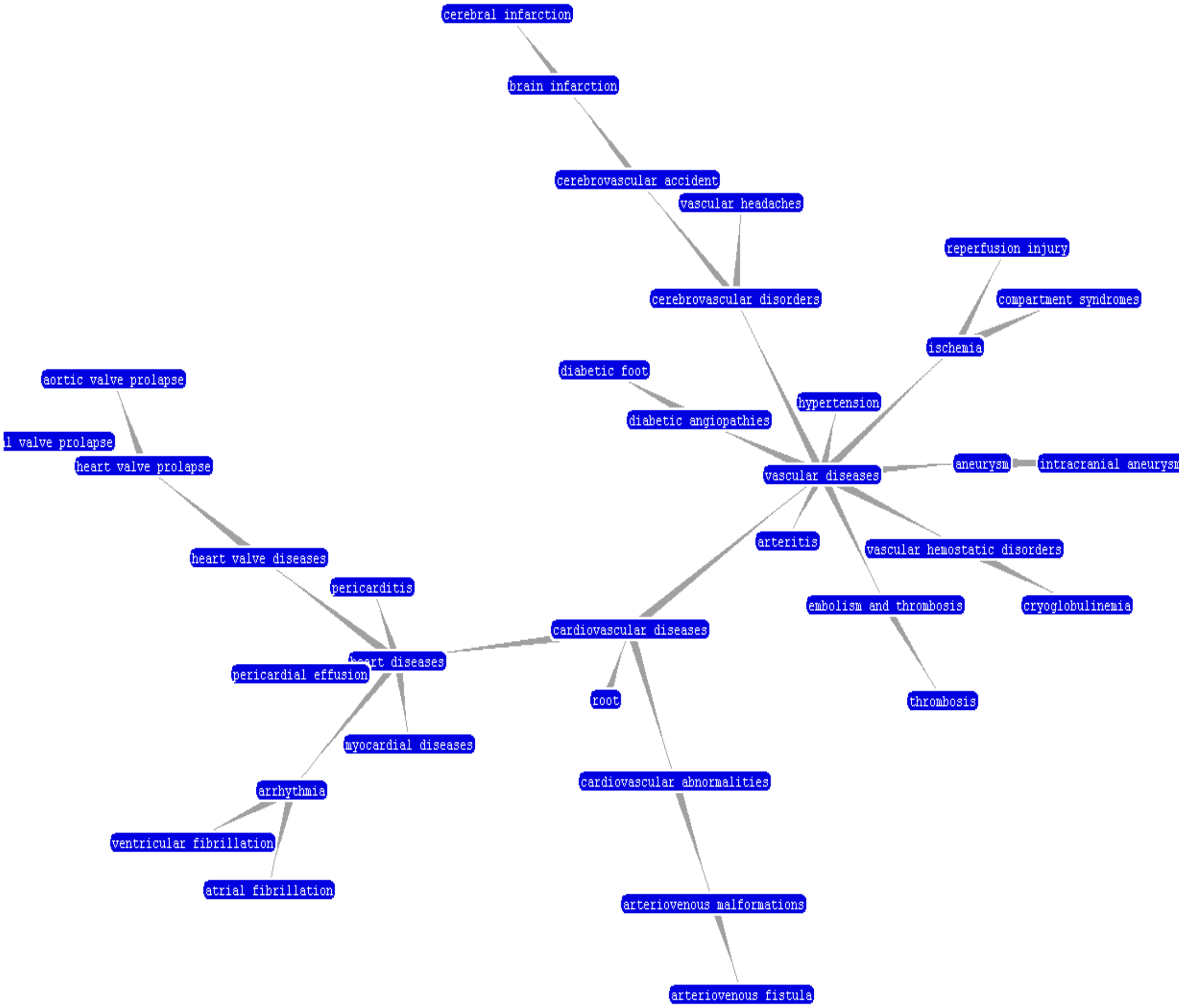

Figure 7: The Gold Standard Taxonomy (MeSH)

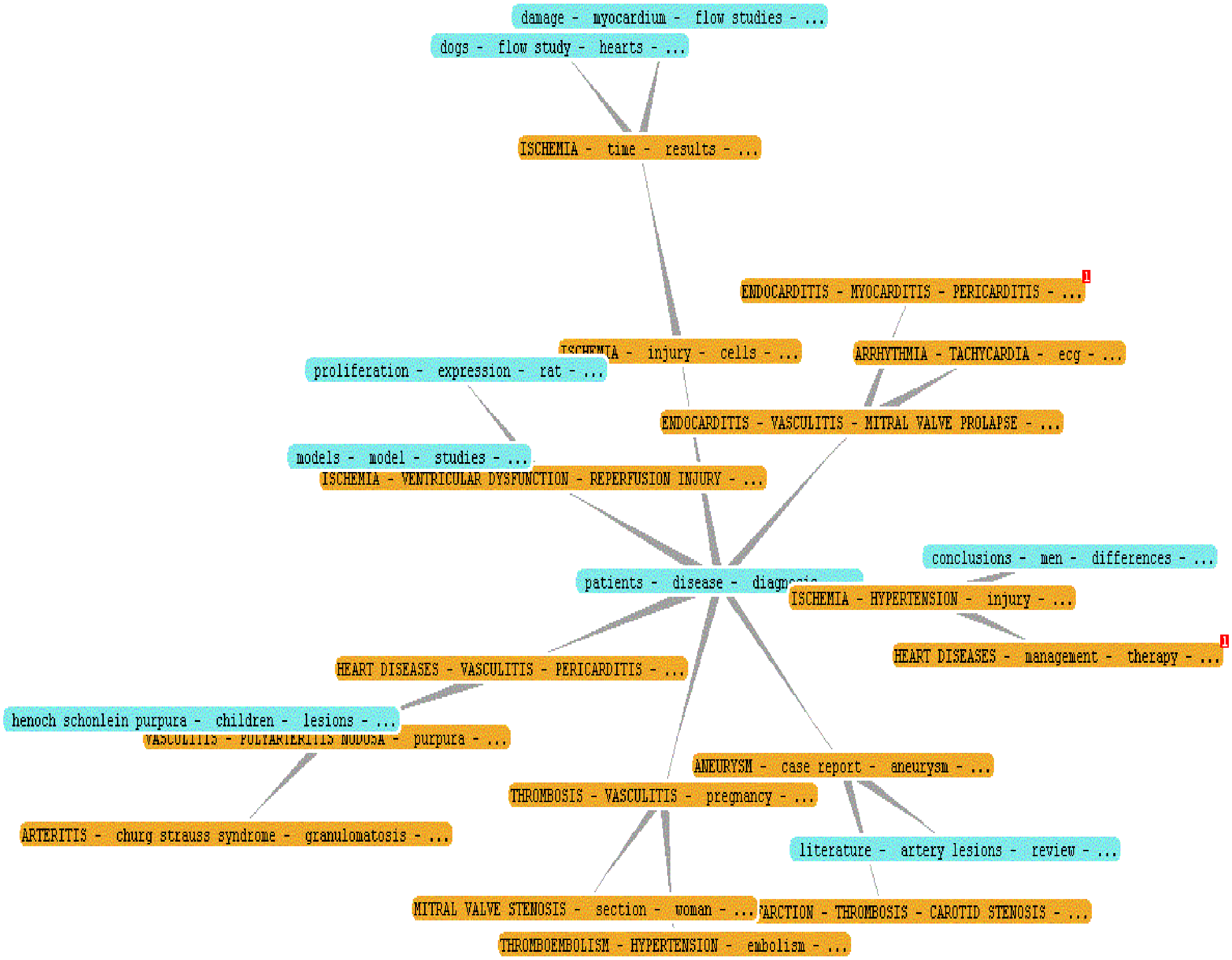
damage - myocardium - flow studies - ...
dogs - flow study - hearts - ...
ISCHEMIA - time - results - ...
ENDOCARDITIS - MYOCARDITIS - PERICARDITIS - ...
1
ISCHEMIA - injury - cells - ...
ARRHYTHMIA - TACHYCARDIA - ecg - ...
proliferation - expression - rat - ...
ENDOCARDITIS - VASCULITIS - MITRAL VALVE PROLAPSE - ...
models - model - studies - ...
ISCHEMIA - VENTRICULAR DYSFUNCTION - REPERFUSION INJURY - ...
conclusions - men - differences - ...
patients - disease - diagnosis
ISCHEMIA - HYPERTENSION - injury - ...
HEART DISEASES - management - therapy - ...
1
HEART DISEASES - VASCULITIS - PERICARDITIS - ...
henoch schonlein purpura - children - lesions - ...
VASCULITIS - POLYARTERITIS NODOSA - purpura - ...
ANEURYSM - case report - aneurysm - ...
THROMBOSIS - VASCULITIS - pregnancy - ...
ARTERITIS - churg strauss syndrome - granulomatosis - ...
literature - artery lesions - review - ...
MITRAL VALVE STENOSIS - section - woman - ...
ARCTION - THROMBOSIS - CAROTID STENOSIS - ...
THROMBOEMBOLISM - HYPERTENSION - embolism - ...

Figure 8: The Learned Taxonomy, Data Set Size = 9305 docs with the matched labels represented in Capital Letters